

# USE OF COMPUTERS TO SOLVE MODEL EQUATIONS

*by S. H. Davis, Jr.*

## *Introduction*

The intimate connection between the complexity of the model equations that can be analyzed and computer availability was pointed out in the previous paper. This warrants spending some time discussing the use of computers to solve different types of transport and rate phenomena problems. This paper will emphasize the aspects of computer utilization which may affect the choice of how (or whether) to solve a particular problem. It will be assumed that we have been given a physical problem and have decided on an adequate mathematical model. We then need to decide what type of computer would be most efficient in solving the model equations. Currently our choice is between the use of a digital or an analog computer, but obviously there is no choice if only one type of machine is on hand. In the future (and currently in a few laboratories) the use of hybrid analog-digital computers may also be feasible.

The characteristics of digital and analog computers that make the use of one type clearly superior over the other in the solution of some problem can be stated rather easily. Digital machines perform arithmetic operations quite accurately and at high speed but in sequence. Analog machines perform arithmetic operations and integration of many variables simultaneously but with limited accuracy and with limited capacity for the size of problem that may be solved. Thus in problems requiring a large number of algebraic operations the use of a digital computer is mandatory. In problems requiring the integration of many differential equations the use of an analog computer has decided advantages. Even in this second case a digital computer could also be used, and the ultimate choice (if offered) is an economic one.

The body of this paper will be devoted to expanding on those factors which are important in determining the feasibility of different types of computer solutions. We shall consider in turn the use of digital and analog computers, and then show how advantages of both types of computers may be combined in a hybrid unit. Finally, a few examples of digital and analog solutions of transport problems will conclude the paper.

---

Editor's Note: Mr. Davis is Associate Professor of Chemical Engineering at Rice University.

*Digital Computation*

The utility of digital computers has been greatly enhanced by two simultaneous developments: 1) increases in size and speed of available machines, and 2) increases in sophistication of machine programs that permit more "natural" and easier communication links between the user and the machine. The parallel between these two developments is aptly described by the computer terms for the objects: "hardware" and "software." Unfortunately, even with the software that has been developed it is still necessary to understand something about how computers work to use them effectively.

First, the importance of sequential operations in a digital computer should be recognized. With few exceptions a digital computer performs only one operation at a time. This means that code and data must be fed to the machine, the machine "told" in what order to start doing arithmetic and logical operations, and data read back out in some predetermined order. Furthermore, the time required for the solution of any problem is proportional (in a general sense) to the number of these operations.

Second, it should be noted that every move that is to be made during the solution of a problem has to be prescribed exactly. Many "obvious" decisions that an engineer would make in the solution to a problem are not a part of the natural intelligence of a computer. If a root of an equation is sought by the time honored "trial and error method," exactly what action is to be taken when a trial is completed must be known. When to terminate such procedures and what action to take if it is decided that the error is too large must both be spelled out. Frequently of great importance (and not often anticipated by the engineer) is the fact that the computer must be told how to decide whether a specified procedure has worked or not, and what to do if for some reason the procedure has not worked.

We will next consider some of the most common general types of problems that arise in the solution of model equations for physical systems. The three types that will be discussed have been broken down according to how much initial numerical analysis needs to be performed before the problem is ready for digital solution.

The first class of problems consists of those that are initially specified in terms of algebraic equations. An example might be the behavior of a collection of bacteria under assumed steady state conditions (i. e., there are no changes with time in the physical system analyzed). There generally are known amounts of different materials put into such a system and (possibly unknown) amounts of material removed, but the balance equations state that the net result is nil for each conserved entity. The problem is that the balance equations usually give information about the conditions in the system in implicit form. Thus, if  $x_1$ ,  $x_2$ , and  $x_3$  describe

the population of bacteria, the oxygen concentration, and the food concentration, respectively, the equations that relate these are usually of the form:

$$f_1(x_1, x_2, x_3, \alpha) = 0 \quad (1)$$

$$f_2(x_1, x_2, x_3, \beta) = 0 \quad (2)$$

$$f_3(x_1, x_2, x_3, \gamma) = 0 \quad (3)$$

where each equation describes a balance for some pertinent quantity. The symbols  $\alpha$ ,  $\beta$ , and  $\gamma$  are included to indicate that there are generally problem parameters that may be different in different circumstances or that might not be known precisely when the problem is first being analyzed. Indeed, one of the purposes of the analysis might be to determine the values of unknown parameters to make experimental data coincide with model predictions.

The form of the equations to be solved (1-3) is usually so complex that no explicit solution can be found. Many schemes have been devised for finding solutions of equations such as these but nearly all are based on one of two techniques:

1) Each function  $f_i(x_1, x_2, x_3, \alpha)$  is split into two parts of the form:

$$f_i(x_1, x_2, x_3, \alpha) = g_i(x_1, x_2, x_3, \alpha) + h(x_1, x_2, x_3, \alpha) \quad (4)$$

such that the first part ( $g_i$ ) of each is simple enough to solve explicitly for the unknowns  $x_1, x_2, x_3$ , that appear in them. As an example:  $g_1 = x_1$ ,  $g_2 = x_2$ ,  $g_3 = x_3$ . Then, assumed values of the unknowns ( $x_1^0, x_2^0, x_3^0$ ) are used to estimate the remaining part of the functions. We then proceed to solve for the unknowns  $x_1, x_2, x_3$  in:

$$g_1(x_1, x_2, x_3, \alpha) = -h_1(x_1^0, x_2^0, x_3^0, \alpha) \quad (5)$$

$$g_2(x_1, x_2, x_3, \beta) = -h_2(x_1^0, x_2^0, x_3^0, \beta) \quad (6)$$

$$g_3(x_1, x_2, x_3, \gamma) = -h_3(x_1^0, x_2^0, x_3^0, \gamma) \quad (7)$$

This gives better values (we hope) for the unknowns to get a better estimate for each  $h$ . Thus, repeated computation of the  $h$ 's and solution for the  $x$ 's are expected to give better and better values for the unknowns. The procedure is terminated usually when the differences between successive values of the  $x$ 's are sufficiently small.

2) The second general method for solving the equations involves defining an overall error associated with a particular guess for the unknowns. For example, if we guess that  $x_1 = x_1^{(k)}$ ,  $x_2 = x_2^{(k)}$ ,  $x_3 = x_3^{(k)}$  then

$$E^{(k)} = [f_1(x_1^{(k)}, x_2^{(k)}, x_3^{(k)}, \alpha)]^2 + [f_2(x_1^{(k)}, x_2^{(k)}, x_3^{(k)}, \beta)]^2 + [f_3(x_1^{(k)}, x_2^{(k)}, x_3^{(k)}, \gamma)]^2 \quad (8)$$

might be used to measure how far we are from satisfying the three

equations. Then we seek values of  $x_i^{(k)}$  which minimize the error (however it is defined),  $E^{(k)}$ . The search for a minimum value of a function is one of the topics covered in a later paper. It suffices here to indicate that this process also leads to repeated computation of algebraic functions of the unknowns  $x_i$  with the hope that eventually values will be found for them that satisfy the system equations.

The second class of problems consists of those that may be found as solutions to sets of ordinary differential equations. Such problems arise for example in determining how systems change with time. Thus, if the balance equations tell us not that all conserved entities that go into some region come out somewhere else but that the difference accumulates in the region, the balance equations have the form:

$$\frac{dx_1}{dt} = f_1(x_1, x_2, \alpha) \quad (9)$$

$$\frac{dx_2}{dt} = f_2(x_1, x_2, \beta) \quad (10)$$

Normally we also know that the unknowns have specified values at some time  $t_0$ , i. e.:

$$x_1(t_0) = x_1^0 \quad (11)$$

$$x_2(t_0) = x_2^0 \quad (12)$$

In the first class of problems everything in the equations was algebraic and could be computed (i. e., the computer could calculate  $f_1$  if  $x_1$ ,  $x_2$ ,  $x_3$ , and  $\alpha$  are given). In the second class of problems the operation of differentiation is not algebraic so that digital solution in the present form is impossible. Furthermore, digital computers operate only on numbers whereas equations (9-12) imply that we are looking for continuous functions of time. In a digital solution we can generally find values for the unknowns only at discrete instants of time  $t_n$ . This is normally done by seeking values for the  $x$ 's at  $t = t_0 + \Delta t$ ,  $t_0 + 2\Delta t$ , etc. If we designate  $t_k = t_0 + k\Delta t$  and  $x_{ik} = x_i(t_0 + k\Delta t)$  the general form of a numerical approximation to the differential equations may be:

$$x_{1k} = g_1(x_{1k-1}, x_{2k-1}, \alpha) \quad (13)$$

$$x_{2k} = g_2(x_{1k-1}, x_{2k-1}, \beta) \quad (14)$$

Thus, again the digital computer is used to compute changes in a set of unknowns in terms of old values for them. Fortunately, most computers contain software packages that take care of all the details of the numerical solution of sets of ordinary differential equations. All the user needs to do is write his equation in some standard form (such as equations [9-12]) then specify the time step  $\Delta t$ , the initial values for the  $x$ 's, the

algebraic relations (f's), the number of steps to be computed (i. e., when to stop), and print out or store information (i. e., what is to be done with the computed values).

If all of the unknowns are not specified at the same time (or value of the independent variable)  $t_0$  but at two or more times then additional problems arise. One usually proceeds by using trial and error, as in the first problem, to guess the unknown initial values and step-wise compute to the time when the values are specified. The difference between the computed values and the specified value is then used to correct the guessed initial values. Problems of this type are referred to as boundary value problems while those with all known initial conditions are called initial value problems. Generally boundary value problems require considerable care just to insure that a solution will be found and in addition may require a large amount of computer time to find accurate results.

The final type of problem that we will discuss includes any system in which the variables are continuous functions of two or more independent variables. In time dependent systems if there are also continuous changes in the unknowns with position then the variables are generally governed by (sets of) partial differential equations. The time dependent heat equation:

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} \quad (15)$$

is the prototype of such equations. Again we must abandon the search for continuous functions of  $t$  and  $x$  and look for values of the unknown(s) at discrete instants of time  $t_n$  and position  $x_k$ . If we limit our discussion to the usual case  $t_n = t_0 + n\Delta t$ ,  $x_k = x_0 + k\Delta x$  and let  $T(t_n, x_k) = T_{nk}$  then numerical procedures can be found to approximate the differential equation by a set of algebraic equations such as:

$$T_{n+1k} = f_k(T_{nk-1}, T_{nk}, T_{nk+1}, \alpha) \quad (16)$$

Note that this is similar to what was found in the case of ordinary differential equations but now the single unknown  $T$  has been replaced by a whole set of unknowns  $T_{nk}$ . The solution may be carried out in much the same manner however, if the problems are of the initial value type. In boundary value problems in partial differential equations (e. g., steady heat conduction in two spatial dimensions) the general form for the discrete approximations is implicit in the unknowns so that some trial and error scheme is often adopted to find the solution.

Although the procedures outlined above may appear to be straightforward, a number of problems present serious obstacles to their implementation. Some major obstacles are:

1) It may be difficult to force trial and error solutions to converge for the entire range of parameters to be analyzed.

2) The errors committed in replacing differential equations by discrete equations are quite dependent on the step size  $\Delta t$  and/or  $\Delta x$ . (In order to make these errors sufficiently small it may be necessary to use very small steps which would require many repetitions of the iteration computation to cover the range of time desired or may—even worse—increase the number of variables at different points in space beyond the storage capacity of the computer.)

3) Estimation of errors is quite complex so that appropriate step sizes must be set either quite conservatively or found by repeating the same calculations with a range of sizes.

### *Analog Computation*

The backbone of modern electronic analog computers is the high gain amplifier since it allows the operations of addition, integration, and function generation to be performed with relatively small errors. In most machines multiplication of variables is performed using function generation elements so that this operation also depends on amplifiers. The ability to integrate continuously is a prime advantage of analog machines. The requirement of an amplifier for each algebraic operation is the major stumbling block to the use of analogs for the solution of complex problems.

Of the three types of problems discussed in the digital section of this paper, analog machines have found their greatest use in solving the second type. This is not surprising since the whole construction of analogs is based on their use as differential equation solvers. Even in the solution of sets of ordinary differential equations, problems may arise if the equations are the boundary value type. Then trial and error methods must be used to find correct initial conditions for the problems. The ability of modern analogs to perform very rapidly and repetitively so that the solution of a given problem may be repeated many times a second allows the solution for the change in one variable to be displayed on a scope; thus, the search for proper initial conditions can be done visually in rapid order.

### *Hybrid Computation*

The ability of analog computers to perform continuous integration of sets of ordinary differential equations very rapidly is a feat that cannot be matched by even the fastest digital computer. The analog is severely limited, however, in the type and number of algebraic and logical operations that it can perform, and in the accuracy of these operations. Thus, there is considerable effort being expended to develop hybrid machines



which will utilize the best features of both types. Such machines may be ideal for solving problems in which a search for best system parameters to match experimental values or for optimum conditions in a system must cover wide ranges of conditions. In such problems many solutions to the same set of differential equations must be performed before the desired conditions are approached. If these can be done using analog techniques and only the final few integrations performed numerically, a great saving of time may be possible.

### *Examples of Computers Used to Analyze Reaction Scheme*

The simplified Krebs Cycle was chosen for a brief study using both an analog and a digital computer to illustrate some properties of both. The purpose of the study was to determine how a cyclic reaction scheme recovers from sudden changes in external conditions. The eight intermediate components in the cycle must reach steady state values that depend on external (to the cycle) conditions such as  $I_1$  supply or  $R_N$  to  $O_N$  reactions. After changes in any of these conditions some time must elapse before the cycle achieves a new steady state level.

The dynamics of chemical reactions are readily simulated using an analog computer. An EAI TR48 was used in this study. Preliminary studies of the behavior of small cycles similar to the eight-member one showed that changes in the cycle occur on two quite different time scales. The slowest changes are those associated with the establishment of a gross balance in the total amount of intermediates present. This is just a balance between production ( $r_s$  in Figure 8 of the paper by Deans) and loss ( $MI_s$ ). More rapid changes occur in establishing the ratios of the different  $I$ 's in the cycle. Since the first time constant is readily computed for any proposed  $r_s$  and  $M$  the rest of the study was devoted to determining how rapidly the components adjust their ratios. This second phase of the analog study offers a good example of what can be done easily with a small analog and what cannot because of size limitations. The dynamics of the  $I_n$  were assumed to be governed by a set of ordinary differential equations of the form:

$$\frac{d I_n}{d t} = r_n(I_{n-1}, I_n) - r_{n+1}(I_n, I_{n+1}) \quad (17)$$

where  $n-1$  and  $n+1$  must be interpreted as components before and after  $n$  in the cycle. All other conditions (such as  $R_N$ ,  $O_N$ ,  $I_1$ ) were assumed constant. The  $I_9$  differential equation was replaced by the algebraic relation:

$$\sum_{n=z}^9 I_n = I \quad (18)$$

on the grounds that the process of achieving equilibrium with respect to loss and production of total I occurs so slowly that the total I is fixed.

The form of  $r_n$  was then chosen as:

$$r_n = f I_{n-1} - b I_n \quad (19)$$

Actually a different forward rate constant  $f$  and backward constant  $b$  should be associated with each reaction  $n$ , but lack of experimental evidence for these dictated some choice and all being equal is the simplest. The complete dynamics of the ring could then be specified by a single constant  $\alpha = b/f$  since the equations may be written:

$$\frac{d y_n}{d \tau} = y_{n-1} - (1 + \alpha) y_n + \alpha y_{n+1} \quad n = 2, 3, \dots, 8 \quad (20)$$

where  $y_n = I_n/I$  and  $\tau = ft$ . The algebraic relation is further:

$$\sum_{n=2}^9 y_n = 1 \quad (21)$$

Each of the seven differential equations requires use of an integrator in an analog simulation. The sum of  $y_n$  in the algebraic relation may be found using 1 to 3 summers. Thus the dynamics of the ring may be studied with a relatively small analog computer. Only a quarter of the amplifiers on the TR48 were used in this part of the simulation. If we wished to modify the reaction model so that changes in  $R_N$  and  $O_N$  occur as a result of other reactions (as was done in studies with small three- and four-member rings) we find that the capacity of a 48 amplifier analog is easily reached. This limitation is set by the number of multipliers available (five on the computer used) to account for reactions which may proceed at rates like:

$$r_7 = f I_6 O_N - b I_7 R_N \quad (22)$$

At best, simulation of such a reaction rate requires one multiplier and two additional amplifiers besides that used in multiplication.

A second capacity limitation was found when it was decided to use as a measure of the deviation in ring ratios from equilibrium the sum:

$$E = \sum_{n=2}^8 \left| y_n - y_{nss} \right| \quad (23)$$

where ss indicates steady state values. Simulation of the differential equations plus this error criterion used up 44 amplifiers. Table I shows the times found, using this circuit, for  $E$  to be reduced to 10% and to 1% of its initial value after an initial condition with all I zero except  $I_2=1$ .



TABLE I

$\alpha = b/f$	$\tau$ at $E = 0.1 E_0$	$\tau$ at $E = 0.01 E_0$
0	6.89	9.09
0.5	2.86	7.70
1.0	3.23	7.93

The analytical solution to the set of equations:

$$\frac{dy_n}{d\tau} = y_{n-1} - (1 + \alpha)y_n + \alpha y_{n+1} \quad n = 2, 3, \dots, 8$$

is of the general form: Plus  $\sum_{n=2}^9 y_n = 1.0$  (24)

$$y_n(\tau) = y_{nss} + \sum_{k=1}^7 a_{nk} e^{\lambda_k \tau} \quad (25)$$

where the  $\lambda_k$  are roots of a polynomial associated with the differential equations. Generally all of the  $\lambda_k$  have negative real parts and thus all the exponentials in the series decay with time. The  $\lambda_k$  with smallest real part determines the time required for any disturbance to decay to some very small value. A digital program was written for the Rice University computer to (a) determine coefficients of the polynomial for the differential equations, and then (b) to use a routine which computes all the roots of a polynomial to determine the root with smallest real part.

Table II shows some of the results of these calculations.

TABLE II

$\alpha$	$\min_k [\text{Re } \lambda_k] = \gamma$	$\ln(100)/\gamma$
0	-0.293	15.7
0.25	-0.366	12.6
0.50	-0.439	10.5
0.75	-0.513	9.97
1.00	-0.586	7.83

For very long times, variations in  $y_n(\tau)$  would follow:

$$y_n(\tau) = y_{nss} + a_n e^{\gamma \tau} \quad (26)$$

An upper bound on the time required for some error criterion to decay to say 1% of its initial value may thus be found as:

$$\tau = \ln(E/E_0)/\gamma \quad (27)$$

For  $E/E_0 = 0.01$  this is:

$$\tau = \ln(100)/\gamma \quad (28)$$

The last column in Table II shows these values. Comparison with Table I indicates that these estimates are indeed always larger than were found in the analog study within the accuracy of the analog simulation.